

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平5-158924

(43)公開日 平成5年(1993)6月25日

(51)Int.Cl.<sup>5</sup>

G 0 6 F 15/20

識別記号

5 3 6

5 3 4

庁内整理番号

7343-5L

7343-5L

F I

技術表示箇所

審査請求 未請求 請求項の数 2 (全 29 頁)

(21)出願番号

特願平3-324728

(22)出願日

平成3年(1991)12月9日

(71)出願人 000001889

三洋電機株式会社

大阪府守口市京阪本通 2 丁目18番地

(72)発明者 古和田 孝之

大阪府守口市京阪本通 2 丁目18番地 三洋電機株式会社内

(72)発明者 長沢 喜美男

大阪府守口市京阪本通 2 丁目18番地 三洋電機株式会社内

(72)発明者 畑山 佳紀

大阪府守口市京阪本通 2 丁目18番地 三洋電機株式会社内

(74)代理人 弁理士 西野 卓嗣

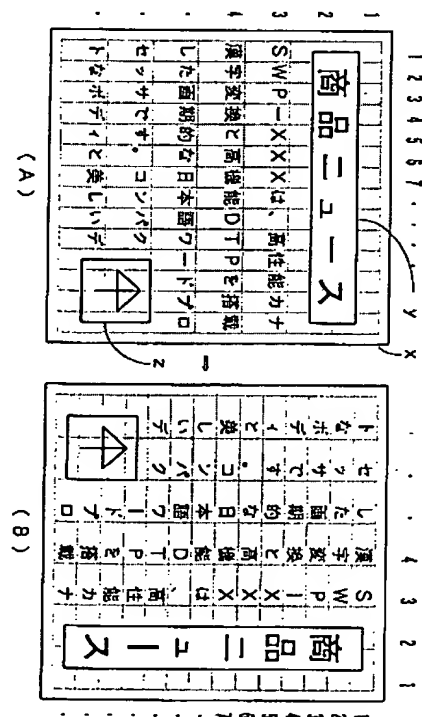
最終頁に続く

(54)【発明の名称】 文書編集装置及び文書編集方法

(57)【要約】

【目的】 本発明は、第25図(A)の文書全体を第25図(B)のように変更する場合、メニュー等で、領域xと領域yのそれぞれの文字列展開方向を横書きから縦書きに変更する操作を簡素化した文字編集装置を提供することを目的とする。

【構成】 本発明は、親領域内のテキスト文字列の配列方向(横書き/縦書き)が変更されたときに、子領域に展開される文字列の配列方向も変更する。



## 【特許請求の範囲】

【請求項1】 文字列を含む複数の文書領域を重ね合わせて編集する文書編集方法において、第1領域を位置的に完全に内包し、且つこの第1領域より下層にある第2領域を前記第1領域の親領域とし、前記第1領域を前記第2領域の子領域とし、前記親領域内のテキスト文字列の展開方向が変更されたときに、前記子領域に展開される文字列の展開方向を自動的に変更する文書編集方法。

【請求項2】 文字列および領域管理情報を入力する入力部(1)と、この入力部(1)より入力された各領域の文字列を格納する領域内データ処理部(2)と、文字列が展開される領域の位置情報および重なり情報と該領域に展開される文字列の展開方向を含む情報を格納する領域管理データ処理部(3)と、この領域と前記文字列を可視的に出力する出力部(4, 5)と、第1領域を位置的に完全に内包し、且つこの第1領域より下層にある第2領域を前記第1領域の親領域とし、前記第1領域を前記第2領域の子領域とし、前記親領域内のテキスト文字列の展開方向が変更されたときに、前記子領域の展開方向の情報を変更して前記領域管理データ処理部(3)に格納させる制御部(6)と、を備える文書編集装置。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】本発明は、ワードプロセッサ等の文書編集装置に関する。特に、DTP機能を備える日本語ワードプロセッサに関する。

## 【0002】

【従来の技術】ワープロセッサ及びワードプロセッサソフトが実行されているパーソナルコンピュータの機能の一つにDTP機能がある。このDTP機能付きのワードプロセッサについて、簡単に説明する。このDTP機能付きのワードプロセッサは、大体において「レイアウト編集機能」と、「WYSIWYG表示機能」を備えている。

【0003】「レイアウト編集機能」によれば、ページの好きな場所に好きな大きさで、矩形の図形・文書領域を設定できる。そして、この領域が文書領域であれば、領域毎の文書の文字列の方向や行ピッチ、文字ピッチ、文字の大きさ、書体などが指定できる。そして、この指定された情報にしたがって文字列データを表示・印刷する。

【0004】この「レイアウト編集機能」では、ページの絶対位置に予め矩形領域を設定しておき、この矩形領域の属性として、文字列の方向や行ピッチ、文字ピッチ、文字の大きさ、書体などが設定・変更できるようになっている。さらに矩形領域の絶対位置や大きさも自由

に変更できるようになっている。これらの仕組みを利用すれば、例えば、図1(A)のように小さめ文字の文章中に大きめ文字の文章を埋め込んだり、図1(B)のように横書き文章中に縦書き文章を埋め込んだりすることが容易にできる。

【0005】尚、この領域としては、図形データ領域・テキストデータ領域等のデータ領域がある。また、この領域は、複数のデータ領域を重ね合わせることができる。また、「WYSIWYG表示機能」によれば、WYSIWYG(WHAT YOU SEE IS WHAT YOU GET)の言葉通り、「見たままが得られる」ということで、印刷結果通りの画面表示が得られる。

【0006】ところで、従来のレイアウト編集機能によれば、まず頁の大きさを設定し、このページの好きな場所に好きな大きさで、矩形の領域を設定していた。そして、この矩形領域毎に文字列の展開方向(縦書/横書)や行ピッチ、文字ピッチ、文字の大きさ、書体などが指定でき、指定された情報にしたがって文字列データを表示・印刷する。

【0007】この矩形領域は、当然、特公平成1-26109号公報(G06F15/20)にも示されるように、設定される位置の情報を備えている。従来、この位置情報は、XY座標情報を基礎に設定される。例えば、A4縦用紙であれば、X座標は「A4縦用紙の左上端から右方向に20mm」、Y座標は、「A4縦用紙の左上端から下方向に30mm」のように、設定される。

【0008】つまり、位置情報をページの絶対位置で個々に保持していた。このため、例えば、図2(A)に示される如く、x領域(下地)の中でy領域をバランスの取れた位置に設定した後に、バランスの取れた状態で、この文書を拡大する場合を考える。この場合、領域(x)を拡大し、さらに領域(y)を拡大したうえで、さらに領域(y)の位置を調整する。

【0009】ところで、例えば第25図(A)の文書全体を第25図(B)のようにしたい場合、メニュー等で、領域xと領域yのそれぞれの文字列方向を横書きから縦書きに変更したうえで、矩形領域yの位置とサイズを変更する必要が生じ、特に、重なる矩形領域の個数が多い時には大変な労力を要した。

## 【0010】

【発明が解決しようとする課題】本発明は、第25図(A)の文書全体を第25図(B)のようにしたい場合、メニュー等で、領域xと領域yのそれぞれの文字列展開方向を横書きから縦書きに変更する操作を簡素化した文字編集装置を提供することを目的とする。

## 【0011】

【課題を解決するための手段】本発明は、親領域内のテキスト文字列の配列方向情報が変更されたときに、子領域に展開される文字列の配列方向情報も変更する文書編集方法である。

## 【0012】

【作用】本発明では、親領域内のテキスト文字列の展開方向が変更されたときに、子領域に展開される文字列の展開方向を自動的に変更する。

## 【0013】

【実施例】図2～図30を参照しつつ、本発明の1実施例を説明する。図3を参照しつつ、この実施例の回路ブロックを説明する。(1)は、文字入力キーを含む複数のキーを有するキーボードよりなる入力部である。この入力部は、文字列および領域管理情報を入力する。

【0014】(2)は、テキスト(文章)や図形などの矩形領域内のテキストデータ・図形データを記憶処理する領域内データ処理部である。この領域内データ処理部

(2)は、入力部(1)より入力された各領域の文字列を格納する。(3)は、矩形領域に関するデータを処理する領域管理データ処理部である。この領域管理データ処理部(3)は、文字列が展開される矩形領域の相対位置情報および重なり情報と該矩形領域に展開される文字列の少なくとも展開方向を含む配置情報を格納する。

【0015】(4)は、プリンタよりなる印刷出力部である。(5)は、LCD表示装置よりなる表示出力部である。(6)は、装置全体の制御を行なう制御部である。又、この制御部(6)は、少なくとも、親領域内のテキスト文字列の展開方向が変更されたときに、子領域の展開方向を変更して領域管理データ処理部(3)に格納させる。

【0016】図4は、このワープロの表示出力部(5)の画面(D)と、入力部(1)の一部を示す図である。図4において、(K1, K2, ……)は、ソフトキー1, 2, ……である。このソフトキー(K1, K2, ……)の有効時、それぞれの機能が画面(D)に表示される。尚、入力部(1)の他のキーは、図示省略した。又、この画面(D)は、WYSIWYG表示状態である。また、(M)は、カーソル位置を示している。

【0017】① まず、領域を管理するためのデータについて説明する。図5に、矩形領域を管理するために領域管理データ処理部(3)に格納される各種データの意味するところを示し、図6に領域管理テーブルを示す。この領域管理テーブルは、複数(領域0～領域5.0)の領域管理データからなる。

【0018】領域管理データは、存在する個々の領域に対応している。図7に一つの領域管理データの内容を示す。この領域管理データは、領域ID番号、領域タイプ、親領域ID番号、領域始点(行位置・桁位置・行端数・桁端数)、領域終点(行位置・桁位置・行端数・桁端数)、領域内データアドレス、上下余白、左右余白、テキスト行数、テキスト桁数、テキスト行ピッチ、テキスト桁ピッチ、テキスト方向、書体およびポイント数から構成される。

【0019】この個々のデータについて、簡単に説明す

る。領域ID番号は、個々の領域を識別するための識別番号である。領域タイプには、テキストと図形の2種類がある。親領域ID番号には、親領域の領域ID番号を保持する(親領域については後で詳細を説明する)。

【0020】領域始点と領域終点には、領域の対角2点の位置が親領域の行桁位置で保持されている。行端数は16分の1行単位を、桁端数は16分の1桁単位の端数を表わし、行と行や桁と桁の間の位置が表わせるようになっている。領域内データアドレスは、領域内に展開すべき領域内データ(図示せず)へのポインタである。

【0021】余白とは、領域最外郭と領域内データを展開する領域有効エリアとの隙間の大きさである。この余白のうち、上下余白には、上と下の隙間の大きさが保持される。左右余白には、左と右の隙間の大きさが保持される。但し後述する縦方向のテキスト領域では上下余白には左と右の隙間の大きさが、左右余白には上と下の隙間の大きさが保持される。

【0022】テキスト行数、テキスト桁数、テキスト行ピッチ、テキスト桁ピッチ、テキスト方向、書体およびポイント数は、領域タイプがテキストのときのみ有効なデータであり、それぞれ領域有効エリア内での行数、桁数、行ピッチ、桁ピッチ、テキストの展開方向、書体およびポイント数を示す。テキスト方向には横と縦があり、横方向テキストでは行は上から下に、桁は左から右に増える。縦方向テキストでは行は右から左、桁は上から下に増える。

【0023】② 次に、領域の重ね合わせについて述べる。この各領域は、位置的に重ね合わせることができる。この重ね合わせについて、図8～図10を参照しつつ説明する。図8は重ね合わせの概念図である。時間的に後から作成あるいは編集した領域が上層に配置されるように重ね合わせる。

【0024】また領域の表示属性として透過／不透過がある。不透過領域は、より下層にある領域に対してクリッピングを行なって重なり部分を表示せずに下層領域を隠す。そして、透過領域は、より下層にある領域に対してクリッピングを行わずに重なり部分も表示して、下層領域を隠さない。テキスト領域は不透過、図形領域は透過となっている。

【0025】図9(A)は、領域重なり順序テーブルを示している。この領域重なり順序テーブルは、領域管理データ処理部(3)に格納されている。領域重なり順序テーブルは、存在する領域の領域ID番号を重なり順序にしたがって格納しており、ある領域がそれよりも領域重なり順序テーブルの前方(配列番号の小さい位置)にある領域の下層にはないことを示す。

【0026】例えば、図9(B)では領域(y)が少なくとも領域(x)の下層には無いことを示している。この重ね合わされた各領域の、親子関係(主従関係)について、図10を参照しつつ説明する。尚、領域の枠を点線で示し

た個所は、下層であることを示している。

【0027】ある領域(6)を完全に内包し且つそれより下層にあるテキスト領域のうちで、最も上層にある領域をこの領域の親領域(4)と呼ぶ。そして、領域

(6)を領域(4)の子領域と呼ぶ。子領域の領域管理データの親領域ID番号は、親領域の領域ID番号である。親領域および親領域の親領域などを含めて祖先領域と呼ぶ。子領域および子領域の子領域などを含めて子孫領域と呼ぶ。

【0028】図10では、領域(1)と領域(2)と領域(3)の親領域は、領域(0)である。領域(4)と領域(5)の親領域は、領域(1)である。領域(6)の親領域は、領域(4)である。

③ カーソルについて述べる。カーソルには、2つの形状があり、用途と動きを異にする。

【0029】一つは図11(A)のような矩形カーソルである。矩形カーソルは文字の上を移動し文字入力ポイント用に使用する。もう一つは図11(B)のような十字カーソルである。十字カーソルは16分の1行単位および16分の1桁単位で移動し主に領域の位置指定等に使用する。下地テキスト領域作成直後は、矩形カーソルになっており文字入力が可能である。

【0030】カーソル位置は、図12のように、カーソルパッケージで表わされる。カーソル行位置およびカーソル桁位置がテキスト領域相対の行位置と桁位置を示し、カーソル領域ID番号が相対基準となるテキスト領域を示す。カーソルは常にその位置での最上層テキスト領域相対で移動するように設計されている。

【0031】④ 文書の作成について述べる。文書の作成には、新規作成と継続作成がある。継続作成は、既存の文書をフロッピディスクなどの記憶装置から呼び出して修正を行なうものである。ここでは新規作成について述べる。

【0032】文書作成キーとして動作するソフトキーを、押下すると画面には、図13の如きメニュー画面が表示される。ここで、カーソルを「新規作成」の位置に移動させ実行キー(図示せず)を押下すると図14のごとき用紙メニュー画面が表示される。ここで、カーソルを変更したい書式設定項に移動させ、漢字変換キーを押下する毎に書式設定事項が代わり、所望の書式設定事項が表示された時点で、実行キー(図示せず)を押下する。これにより、図4のごとき、所望の書式の表示がなされる。

【0033】この書式設定により、ワープロの内部では、最初の領域の設定が為される。つまり、上記操作により、用紙サイズ、1頁あたりの行数、1行あたりの文字数(桁数)およびテキストの展開方向(縦/横)を指定実行すると文書のベースとなる下地テキスト領域が、領域ID番号が「0」で、領域タイプが「テキスト」の領域として作成される。親領域ID番号としては、この

下地テキスト領域に限り自分自身の領域ID番号である。

【0034】下地テキストデータは、新規作成直後には1行分(桁数は図14のメニューで設定された文字数の値)だけ作られる。従って、下地テキスト領域は(テキスト行数=1、テキスト桁数=メニューで設定された値)となる。領域始点および領域終点には、下地テキスト領域自身の先頭行桁(行桁端数は0)および最終行桁(行桁端数は15/16)が格納される。

【0035】後に、領域内データ編集で下地テキストの行数が増減するに従って領域終点およびテキスト行数の値を更新する。その他の領域管理データにはメニューからの数値(テキスト方向)や、それをもとに計算した結果(テキスト行ピッチ、テキスト桁ピッチ)あるいはデフォルト値(上下余白、左右余白、書体、ポイント数)が入れられる。

【0036】但し、テキスト行ピッチおよびテキスト桁ピッチの値は後で述べる子領域への継承値として利用するためのもので、実際に行ピッチおよび桁ピッチは行毎に設定可能な別途用意している下地テキスト行桁ピッチテーブルに格納されている。また、領域重なり順序テーブルの先頭に領域ID番号0が格納され、下地テキスト領域は以降作られる全ての領域の祖先領域となる。

【0037】⑤領域編集について説明する。領域編集の機能としては、領域作成、領域削除、領域移動、領域サイズ変更および領域属性変更がある。領域作成以外の領域編集を行なう場合は、領域編集キーとして作用する図4のソフトキー(K2)を押下すると図15の如く、領域編集モードになる。

【0038】この押下に伴って矩形カーソルから十字カーソルに変わり行ピッチの16分の1単位および桁ピッチの16分の1単位での移動が可能となる。領域編集を終了するときには、終了キー(図示せず)を押下する。この押下にもなって十字カーソルから矩形カーソルに変わり文字単位で移動するようになる。

【0039】⑤-①領域作成について説明する。領域作成は、領域編集の機能の一つであるが、使い勝手を良くするために、図4に示される如く、領域作成キー(K1)は、領域編集キー(K2)とは別に設けた。新たに領域を作成する場合は、領域作成キーとして作用する図4のソフトキー(K1)を押下する。

【0040】この押下に伴って矩形カーソルから十字カーソルに変わり、この十字カーソルで作成したい領域の対角2点(始点と終点)を指定することにより、新たな領域ID番号を持つ領域が新規作成される。なお、十字カーソル移動中に領域タイプの指定が可能となっている。このとき指定された始点・終点の位置から新規作成される領域の親領域を検出し、領域管理データの親領域ID番号に親領域の領域ID番号を納める。

【0041】さらに指定された始点と終点の位置を親領

域相対の位置に変換し、領域管理データの領域始点及び領域終点に格納する。新規作成された領域が、テキスト領域の場合、テキスト方向、書体、ポイント数、テキスト行ピッチおよびテキスト桁ピッチを親領域から継承し、それぞれ領域管理データに格納する。

【0042】さらに、これらの値と領域のサイズから上下余白、左右余白、テキスト行数およびテキスト桁数を算出し、各々領域管理データに格納する。新規作成された領域が図形領域の場合、上下余白および左右余白にはデフォルトとして0が格納される。さらに領域重なり順序テーブルの最後尾に、新規作成した領域の領域ID番号を格納しておく。

【0043】⑤-②領域削除について説明する。すでに作成されている領域を削除する場合、領域編集モードで十字カーソルにより削除したい領域を指定して領域削除キーとして作用する図15のソフトキー（K10）を押下する。領域を削除する場合、その子孫領域も一緒に削除されるようになっており、これらの領域について領域管理テーブルおよび領域重なり順序テーブルからのデータの削除を行なう。

【0044】⑤-③領域移動について説明する。すでに作成されている領域を移動する場合、領域編集モードで十字カーソルにより移動したい領域を指定して領域移動キーとして作用する図15のソフトキー（K1）を押下し、さらに移動先の始点位置を十字カーソルで指定する。領域を移動する場合、その子孫領域も一緒に移動されるようになっており、これらの領域について領域管理テーブル（親領域、始点、終点など）および領域重なり順序テーブルのデータの更新を行なう。

【0045】⑤-④領域サイズ変更について説明する。すでに作成されている領域のサイズ変更する場合、領域編集モードで十字カーソルにより移動したい領域を指定して領域サイズ変更キーとして作用する図15のソフトキー（K3）を押下し、さらにサイズ変更後の終点位置を十字カーソルで指定する。

【0046】領域のサイズ変更をする場合、その子孫領域も一緒にサイズ変更されるようになっており、これらの領域について領域管理テーブル（親領域、始点・終点、余白、ピッチなど）および領域重なり順序テーブルのデータの更新を行なう。さらに、領域サイズ変更について、図16～図22のフローチャートを参照して詳細を述べる。

【0047】図16は、領域サイズ変更のメインルーチンのフローチャートを示している。図17は、メインルーチンのステップS102のサブルーチンのフローチャートを示している。図18は、図17のサブルーチンのフローチャートを示している。図19は、メインルーチンのステップS106のサブルーチンのフローチャートを示している。

【0048】図20は、メインルーチンのステップS101

のサブルーチンのフローチャートを示している。図21は、メインルーチンのステップS104のサブルーチンのフローチャートを示している。図12は、メインルーチンのステップS103のサブルーチンのフローチャートを示している。

【0049】図16を参照しつつ、説明する。まず、ステップS101でカーソル位置に存在する領域の中で最も上層にある領域の領域ID番号(id)を求め、領域サイズ変更の対象領域とする。ステップS102で対象領域の子孫領域を求め、バッファ(idlist)に対象領域と求めた子孫領域の領域ID番号を格納しておく。

【0050】ステップS103でバッファ(idlist)に格納された領域についてのみの重なり順序関係を、領域重なり順序テーブルから抽出し、セーブバッファ(saveoverlaplist)にセーブしておく。ステップS104でバッファ(idlist)に格納された領域ID番号を領域重なり順序テーブルから削除する。これにより親領域検出時の対象外となり自分自身が親領域になるのを防ぐ。

【0051】ステップS105で対象領域の始点(str)と終点(end)を再設定する。すなわち始点(str)には対象領域の領域管理データの領域始点をそのまま使用し、終点(end)にはユーザが設定した新たな位置を代入する。ステップS106で始点(str)および終点(end)からサイズ変更後の新たな親領域ID番号(pid)を求める。

【0052】ステップS107で始点(str)および終点(end)を親領域(pid)相対の行桁位置に変換する。ステップS108でセーブバッファ(saveoverlaplist)の内容を重なり順序テーブルの最後尾に追加する。ステップS109で対象領域の領域タイプを調べ、図形領域ならステップS111へ、文書領域ならステップS110へ進む。

【0053】ステップS110でセーブバッファ(saveoverlaplist)に格納されている領域のなかで領域タイプが文書のもの全てについて、行ピッチおよび桁ピッチの再計算（テキスト行数およびテキスト桁数固定でテキスト行ピッチおよびテキスト桁ピッチを算出し直す）を行ない、それぞれの領域管理データの値を更新する。ステップS111で領域サイズ変更の処理を終了する。

【0054】図2を参照しつつ、サイズ変更の様子を子領域を持つテキスト領域を拡大した場合について、説明する。

⑤-④-①図2を参照しつつ領域サイズ変更について説明する。図2(A)は変更前の状態を、図2(B)は変更後の状態を示している。図2(A)ではテキスト領域yがテキスト領域xに貼り付いた形になっている。

【0055】領域yの領域始点は（2行、3桁）、領域終点は（6行、7桁）となっている。この状態で領域(x)を拡大すると領域(x)のテキスト行ピッチおよびテキスト桁ピッチが計算し直されて図2(B)のように大きくなる。さらに領域(y)の領域始点および領域終点の値は更新しないため領域(y)の絶対的なサイズは領域(x)

の拡大とともに大きくなる。従って領域(y)に関してもテキスト行ピッチおよびテキスト桁ピッチを計算し直し図2(B)のように大きくする。

【0056】⑤-⑤領域属性変更について説明する。すでに作成されている領域の属性を変更する場合、領域編集モードで十字カーソルにより属性変更したい領域を指定して領域属性変更キーとして作用する図15のソフトキー(K9)を押下する。尚、図4のソフトキー(K3)を押下しても、この領域属性変更のモードには入れ

る。

【0057】領域属性には、図形領域では上下左右の余白、テキスト領域ではそれに加えて領域内での有効行数／桁数、テキストの展開方向、書体およびポイント数があり、これらを設定するための図23のメニューが現われる。メニューでの設定値を元に領域管理データの値を更新する。例えば、テキスト領域の上下余白が変更された場合、領域管理データの上下余白を更新するとともに領域有効エリアに有効行がピッタリ収まるように行ピッチを計算し直す。

【0058】また、テキスト領域の有効行数が変更された場合、領域管理データのテキスト行数を更新するとともに領域有効エリアに有効行がピッタリ収まるように行ピッチを計算し直す。さらにテキスト領域のテキスト展開方向を変えた場合、領域管理データのテキスト方向を更新するとともに領域有効エリアに有効行および有効桁がピッタリ収まるように行ピッチおよび桁ピッチを計算し直す。

【0059】これら領域属性を変更する場合、その子孫領域もその影響を受け、サイズや位置、ピッチ、テキスト方向等が更新される。テキスト方向(縦書／横書)は、親領域が反転(縦／横逆転)されれば子領域も反転する。すなわち親も子も横の場合に、親を縦にすれば子も縦になり、親も子も縦の場合に親を横にすれば子も横になり、親が横で子が縦の場合に親を縦にすれば子は横になり、親が縦で子が横の場合に親を横にすれば子は縦になる。

【0060】⑤-⑤-①フローチャートを参照しつつ縦書／横書の変更に関して説明する。テキストの展開方向(縦書／横書)を変更したときの処理について、図24、図17、図18、図20～図22のフローチャートを参照して詳細を述べる。図24は、テキスト方向変更のメインルーチンのフローチャートを示している。

【0061】図17は、メインルーチンのステップS202のサブルーチンのフローチャートを示している。図18は、図17のサブルーチンのフローチャートを示している。図20は、メインルーチンのステップS201のサブルーチンのフローチャートを示している。

【0062】図21は、メインルーチンのステップS204のサブルーチンのフローチャートを示している。図22は、メインルーチンのステップS203のサブルーチンのフ

ローチャートを示している。図24を参照しつつ、説明する。

【0063】まず、ステップS201でカーソル位置に存在する領域の中で最も上層にある領域の領域ID番号(id)を求め、領域属性変更(テキスト方向変更)の対象領域とする。ステップS202で対象領域の子孫領域を求め、バッファ(idlist)に対象領域と求めた子孫領域の領域ID番号を格納しておく。

【0064】ステップS203でバッファ(idlist)に格納された領域についてのみの重なり順序関係を、領域重なり順序テーブルから抽出し、セーブバッファ(saveoverlaplist)にセーブしておく。ステップS204でバッファ(idlist)に格納された領域ID番号を領域重なり順序テーブルから削除する。

【0065】ステップS205でセーブバッファ(saveoverlaplist)の内容を重なり順序テーブルの最後尾に追加する。ステップS206でセーブバッファ(saveoverlaplist)に格納されている領域のなかで領域タイプが文書のもの全てについて、それぞれの領域管理データのテキスト展開方向を反転(縦／横逆転)させる。

【0066】ステップS207でセーブバッファ(saveoverlaplist)に格納されている領域のなかで領域タイプが文書のもの全てについて、行ピッチおよび桁ピッチの再計算を行ない、それぞれの領域管理データの値を更新する。尚、この再計算とは、テキスト行数およびテキスト桁数固定でテキスト行ピッチおよびテキスト桁ピッチを算出し直すことである。

【0067】ステップS208で領域属性変更(テキスト方向変更)の処理を終了する。

⑤-⑤-②図25を参照しつつ縦書き／横書きの変更に  
 関して説明する。属性変更の様子をテキスト領域のテキスト展開方向を横書きから縦書きに変更した場合を例にして図25に示す。図25(A)は変更前の状態を、図25(B)は変更後の状態を示している。

【0068】図25(A)では横方向のテキスト領域(y)と図形領域(z)が横方向のテキスト領域(x)に貼り付いた形になっている。領域(y)の領域始点は(1+8/16行、1+8/16桁)、領域終点は(1+8/16行、14+8/16桁)である。領域(z)の領域始点は(6行、11+8/16桁)、領域終点は(7+8/16行、14+8/16桁)となっている。

【0069】この状態で領域(x)のテキストの展開方向を横から縦に変更すると、領域(x)に展開されるテキストデータは図25(B)のように右上から左下に向かって展開される。それと同時にテキスト領域(y)は領域始点及び領域終点が親領域(x)相対の行桁位置で示されているため絶対位置が移動する。

【0070】同様に図形領域zの絶対位置も移動する。さらに領域(x)のテキスト方向の反転にともない子孫領域のうちのテキスト領域については、テキスト方向を反転させるとともに、領域有効エリアに有効行および有効

桁がピッタリ収まるように行ピッチおよび桁ピッチを計算し直す。すなわち横方向だった領域(y)のテキスト方向を縦に変更して行ピッチおよび桁ピッチを計算し直す。

【0071】⑥下地テキストによる子領域への影響に関して説明する。下地テキストの挿入／削除操作、及び行ピッチ変更操作等の編集を行なった時の子領域への影響について説明する。下地テキストの編集が直接影響を与えるのは下地テキストの子領域のみであるが、結果的には子孫領域にも影響は及ぶことになる。

【0072】⑥-①下地テキストの行単位での挿入／削除操作について説明する。子領域に影響を与える行の挿入／削除のパターンを図26に示す。図26(A)は、行挿入／削除の対象行よりも子領域の始点が後方にある場合、図26(B)は行挿入／削除の対象行よりも子領域の始点が前方、子領域の終点が後方にある場合である。

【0073】図26(A)のパターンでは行の挿入／削除に伴い子領域の始点および終点の行位置を、増減のあった行数分だけ増減させる。例えば3行挿入されれば図26(C)のように子領域の始点および終点の行位置をそれぞれプラス3する。図26(B)のパターンでは行の挿入／削除に伴い子領域の絶対サイズを保持するよう子領域の終点の行桁位置を更新する。

【0074】例えば3行削除されれば図26(D)のように子領域の絶対サイズを保持したまま子領域の終点を更新する。下地テキスト行削除によって子領域が下地テキストからはみ出す可能性のある場合は、子領域が下地テキスト内に収まるように予め下地テキストの最終行以下に新たな空白行を作成してから行削除処理を行なう。

【0075】⑥-②下地テキストの行単位での行ピッチ変更操作について説明する。子領域に影響を与える行ピッチ変更のパターンは図27(A)のように行ピッチ変更の対象行よりも子領域の始点が前方、子領域の終点が後方にある場合だけである。この状態で行ピッチ変更が行なわれた場合、子領域の始点および終点の更新は行なわない。

【0076】従って、下地テキストの行ピッチの変更にともない子領域の絶対サイズが変わるため、図27

(B)の如く子領域に対してテキスト行数、テキスト桁数、上下余白および左右余白を固定したままテキスト行ピッチおよびテキスト桁ピッチの更新を行なう。⑦用紙変更の処理に関して説明する。

【0077】下地テキスト以外の子領域に関しては、前述の属性変更操作により、その寸法が変更される。用紙変更処理キーとして作用する印刷キー（図示せず）を押下して、図28のメニュー画面を表示する。そして、用紙の縦横、及び、横書縦書の両方を変更して、実行キーを押すと、用紙変更の処理が行われる。

【0078】続いて、再度実行キーを押すと印刷が実行され、この実行キーの押下に代えて取消キー（図示せ

ず）を押すと、図4のモードに復帰する。この領域編集モードに復帰したとき、例えば、文書は図29から図30に変化している。図31にこの用紙変更の内部処理動作の様子を示す。

【0079】図31(A)は、用紙を縦方向に使用し、複数の領域を使ってレイアウトされている。横線でハッチングがかかっているのは、テキスト展開方向が横の領域である。これら以外は、テキスト展開方向が縦の領域である。この文書全体を図31(C)のように、用紙を横方向に使用し、文字列方向を反転（縦横逆点）させたい場合の処理について述べる。

【0080】まず内部処理として、領域属性変更で下地テキスト領域のテキスト展開方向を縦から横に変更する。これにより図25と同様の処理が行なわれ、全てのテキスト領域のテキスト展開方向が反転（縦横逆点）するとともに、見かけ上の領域位置が変化する。この時点では用紙のサイズは変更していないので、内部的には図31(B)のようになっている（行桁ピッチが適当でない）。

【0081】次の内部処理として、領域サイズ変更で下地テキスト領域のサイズを変更する（縦と横のサイズを入れ替える）。これにより、図2と同様の処理が行なわれ、見かけ上の領域位置が変換し、図31(C)のようになる（行桁ピッチが適当な値となる）。

【0082】

【発明の効果】「⑤-⑤-②」の項で一例を説明した如く、本発明では、親領域内のテキスト文字列の展開方向が変更されたときに、子領域に展開される文字列の展開方向を自動的に変更するので、領域の文字列の展開方向の修正作業の軽減が図れ、文書作成中における文書レイアウト修正作業の操作性が向上する。

【図面の簡単な説明】

【図1】従来例を説明するための図である。

【図2】領域サイズ変更を示す図である。

【図3】本発明の一実施例の構成を示すブロック図である。

【図4】画面を示す図である。

【図5】領域に関するデータの意味を示す図である。

【図6】領域管理テーブルの構成図である。

【図7】領域管理データの構成図である。

【図8】領域を重ね合わせた文書の概念図である。

【図9】領域重なり順序テーブルの構成図である。

【図10】領域の主従関係を説明する図である。

【図11】カーソルの形状を示す図である。

【図12】カーソルバケットの構成図である。

【図13】画面を示す図である。

【図14】画面を示す図である。

【図15】画面を示す図である。

【図16】領域サイズの変更のフローチャートを示す図である。

- 【図17】フローチャートを示す図である。  
 【図18】フローチャートを示す図である。  
 【図19】フローチャートを示す図である。  
 【図20】フローチャートを示す図である。  
 【図21】フローチャートを示す図である。  
 【図22】フローチャートを示す図である。  
 【図23】画面を示す図である。  
 【図24】テキスト方向の変更のフローチャートを示す図である。  
 【図25】領域属性変更を示す図である。  
 【図26】下地テキストの行挿入／行削除を示す図である。  
 【図27】下地テキストの行ピッチ変更を示す図である。

\* する。

【図28】画面を示す図である。

【図29】用紙変更前の文書の一例を示す図である。

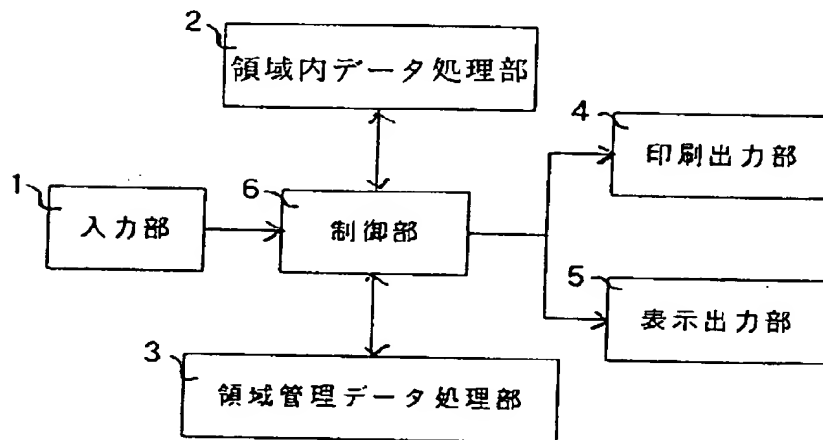
【図30】用紙変更後の文書の一例を示す図である。

【図31】用紙変更による動作を説明するための図である。

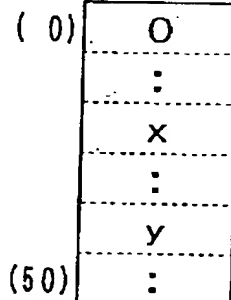
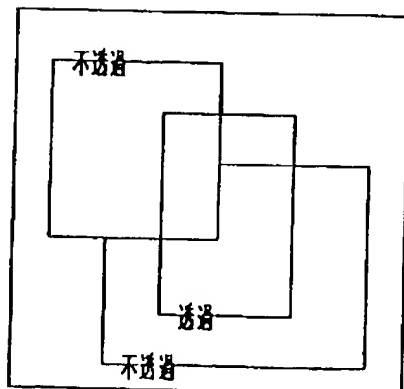
【符号の説明】

- (1) 入力部、  
 (2) 領域内データ処理部、  
 (3) 領域管理データ処理部、  
 (4) 印刷出力部(出力部)、  
 (5) 表示出力部(出力部)、  
 (6) 制御部。

【図3】

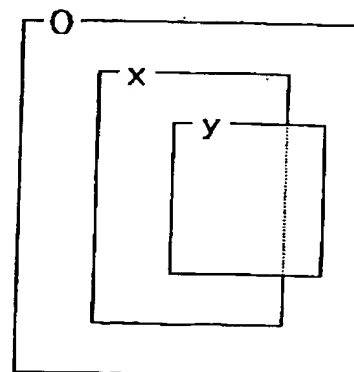


【図8】



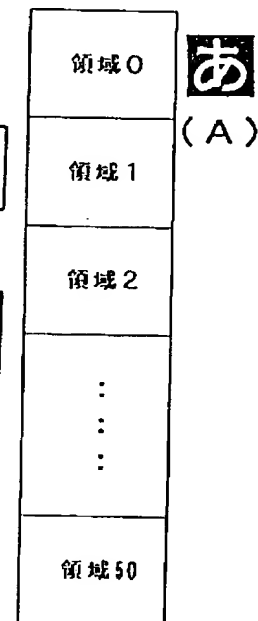
(A)

【図9】

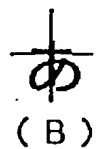


(B)

【図6】

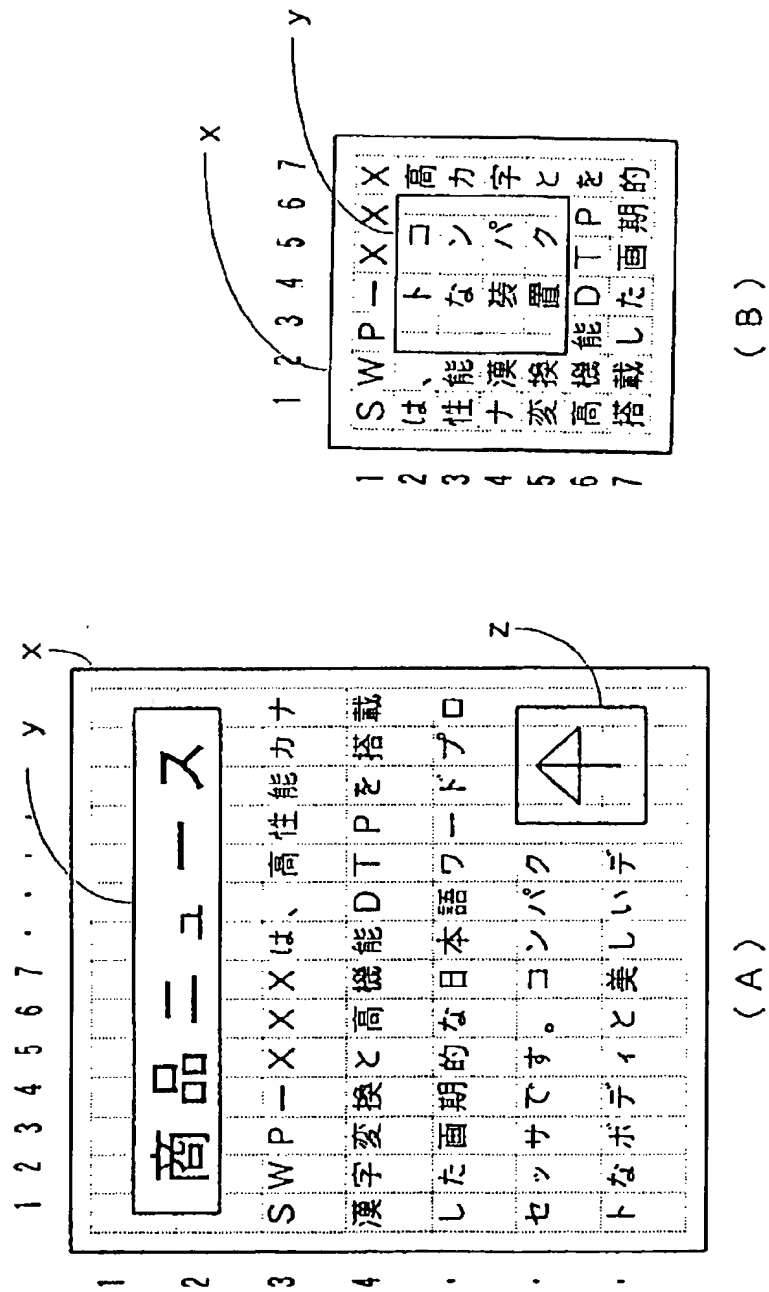


【図11】

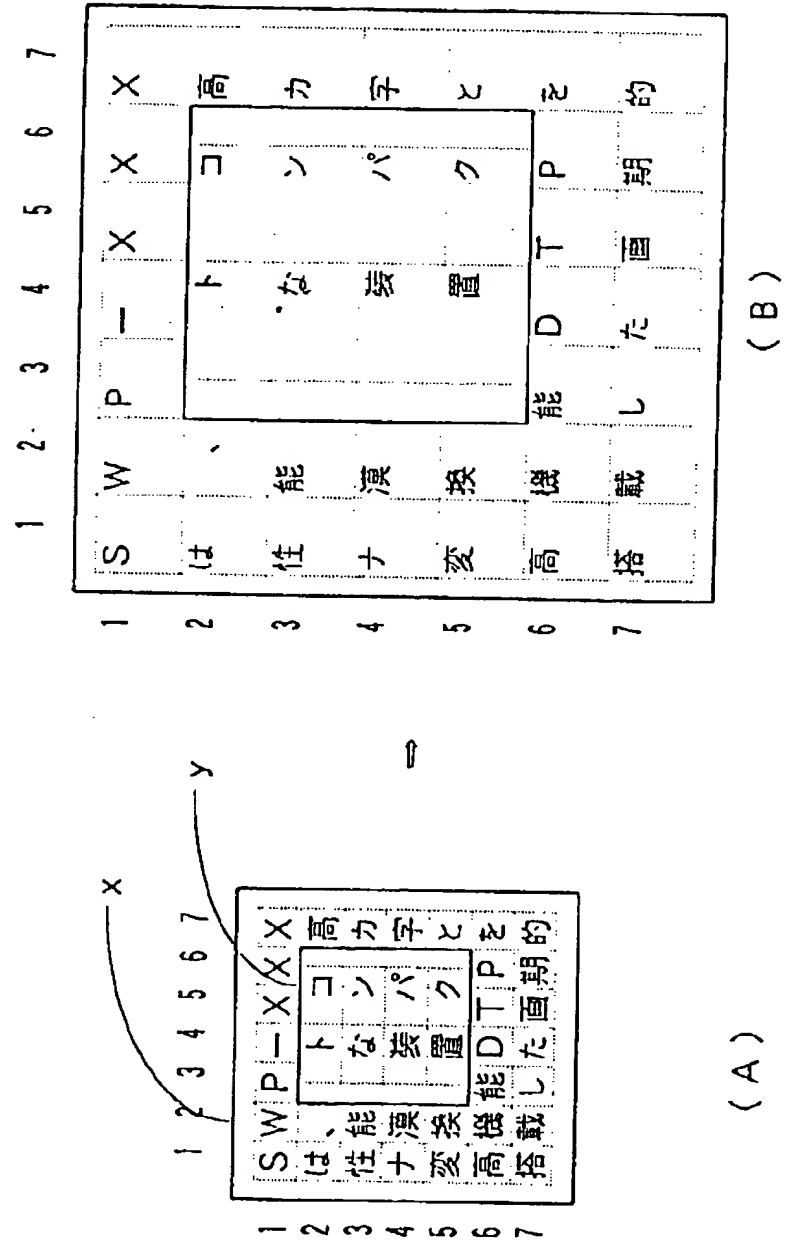




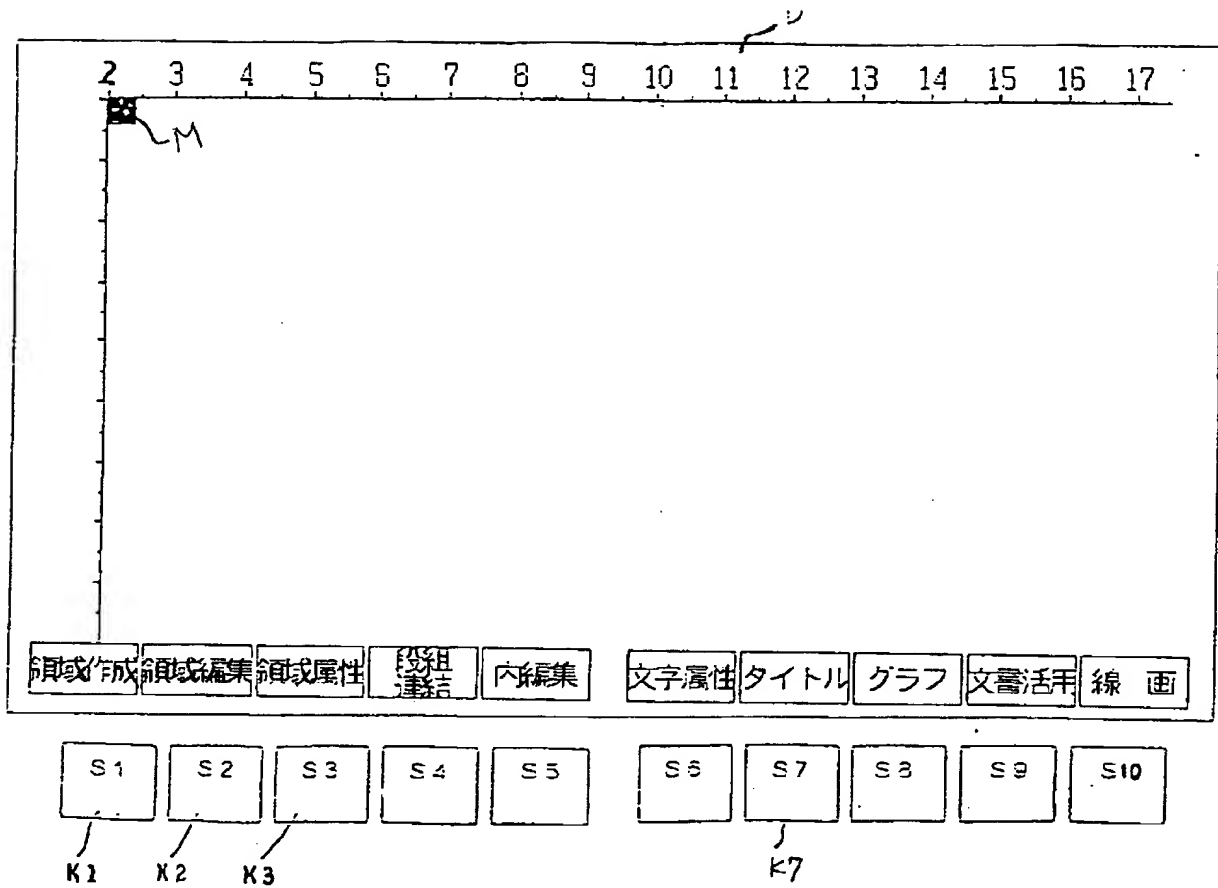
【図1】



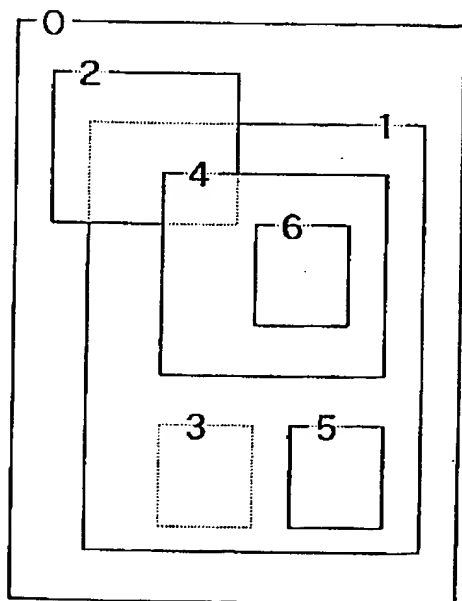
【図2】



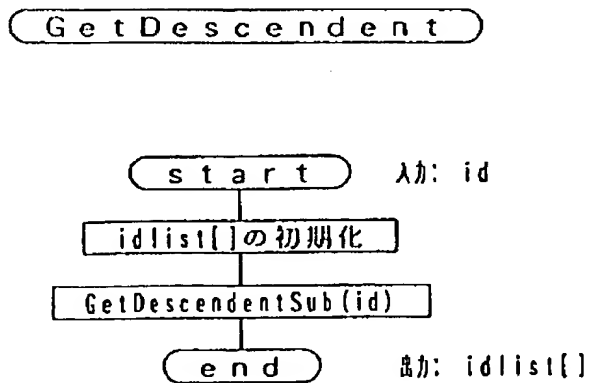
【図4】



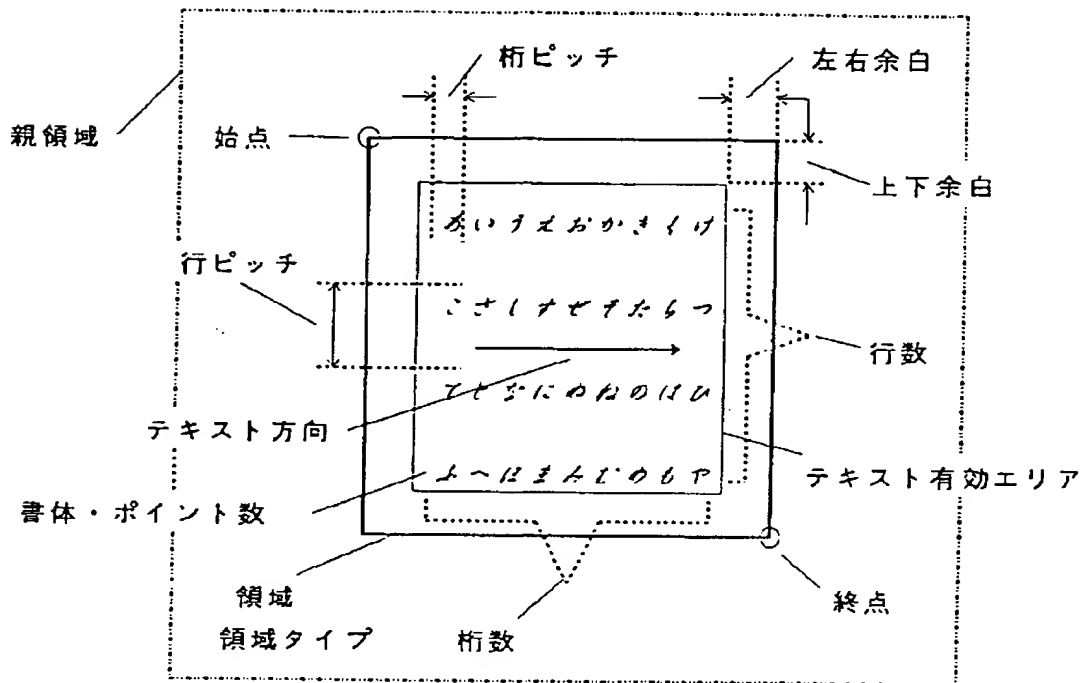
【図10】



【図17】



【図5】



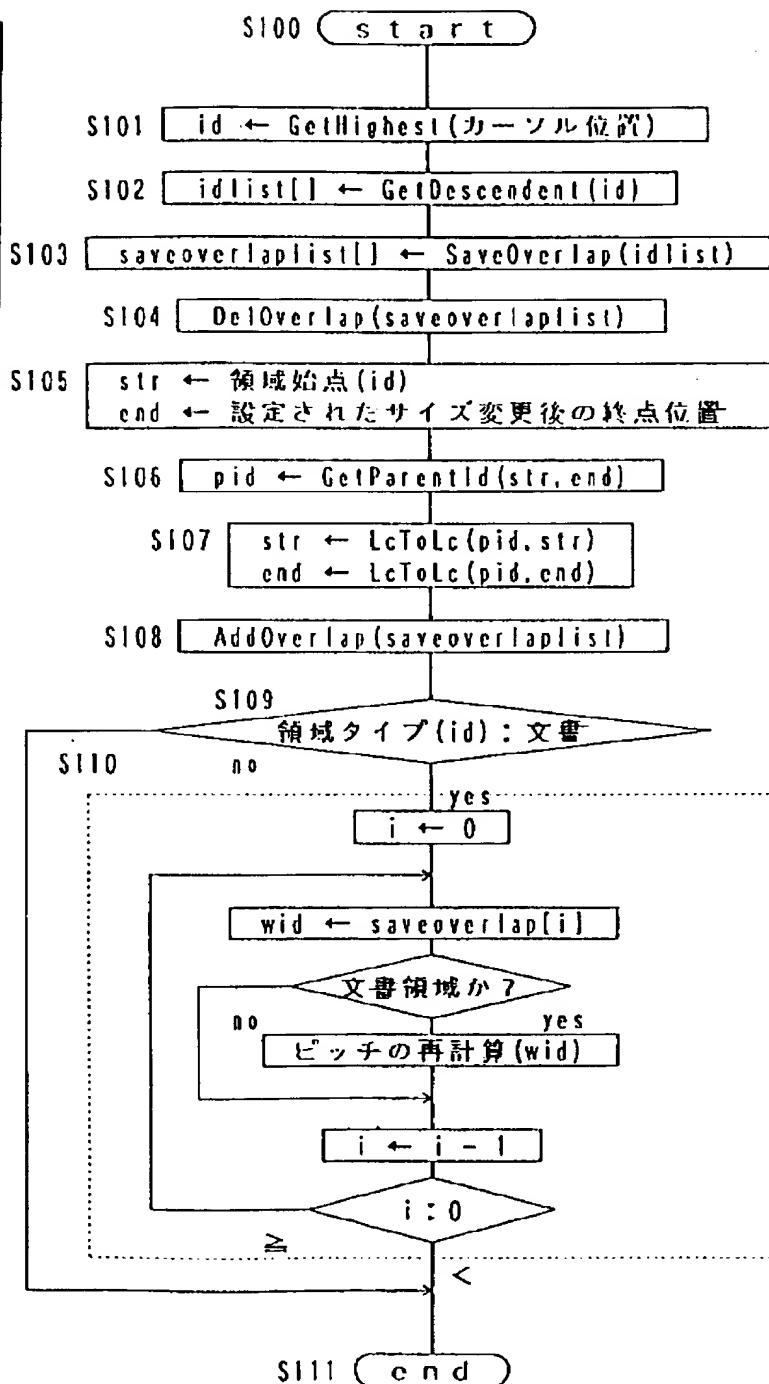
【図12】

カーソル領域   D 番号
カーソル行位置・行
カーソル行位置・行端数
カーソル桁位置・桁
カーソル桁位置・桁端数

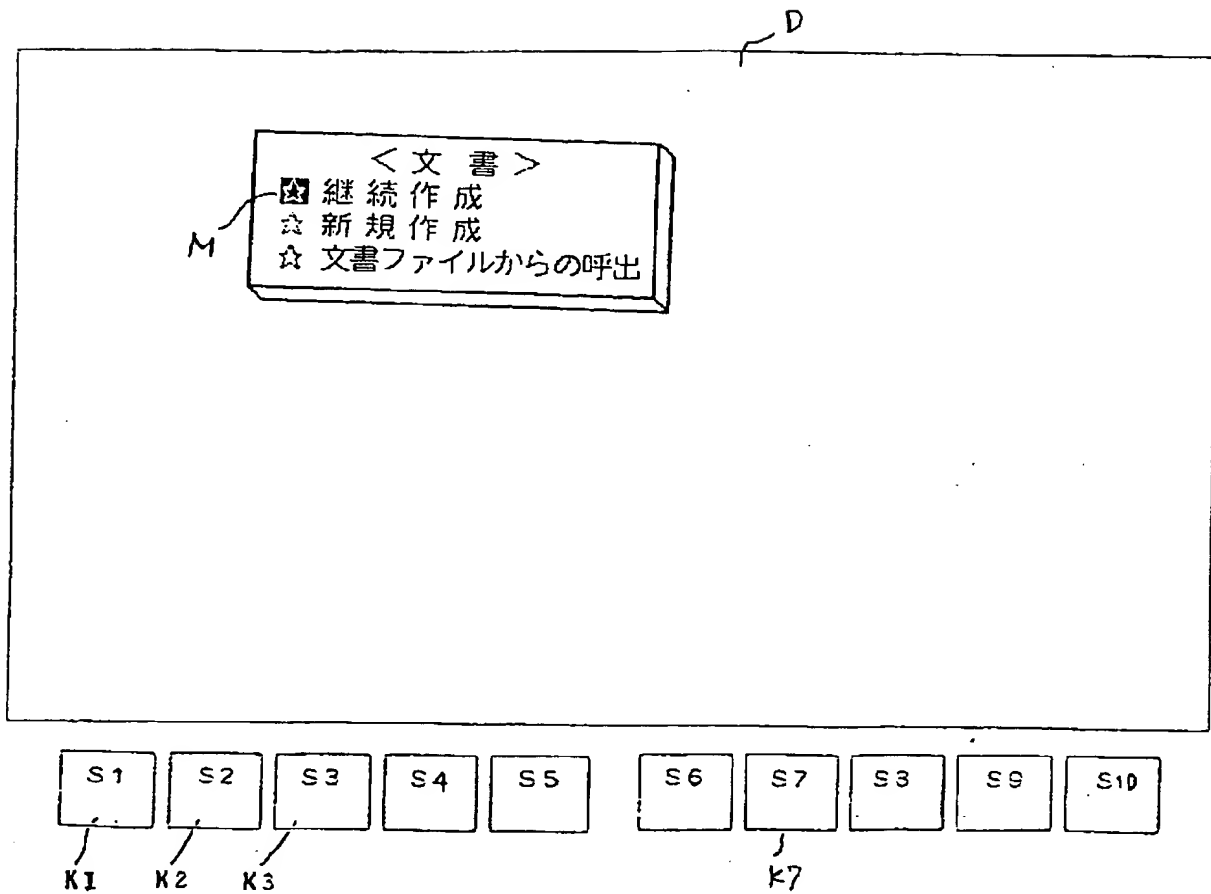
【図7】

領域ID番号
領域タイプ
親領域ID番号
領域始点行位置
〃    桁位置
〃    行端数
〃    桁端数
領域終点行位置
〃    桁位置
〃    行端数
〃    桁端数
領域データアドレス
上下余白
左右余白
テキスト行数
〃    桁数
〃    行ピッチ
〃    桁ピッチ
〃    方向
書体
ポイント数

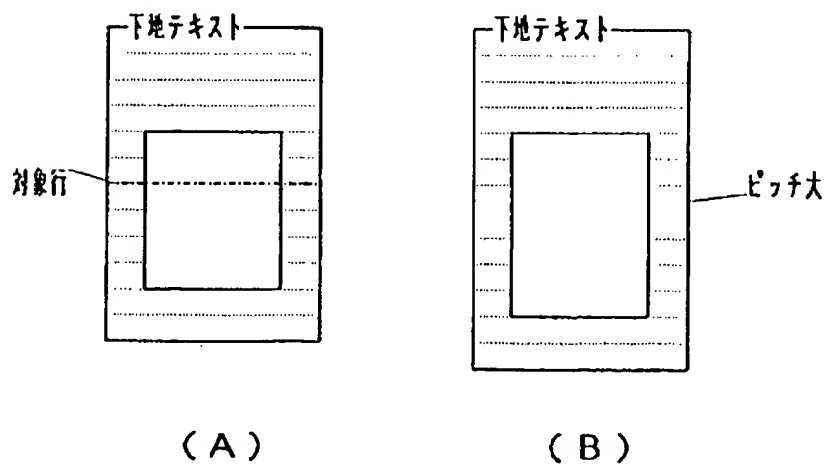
【図16】



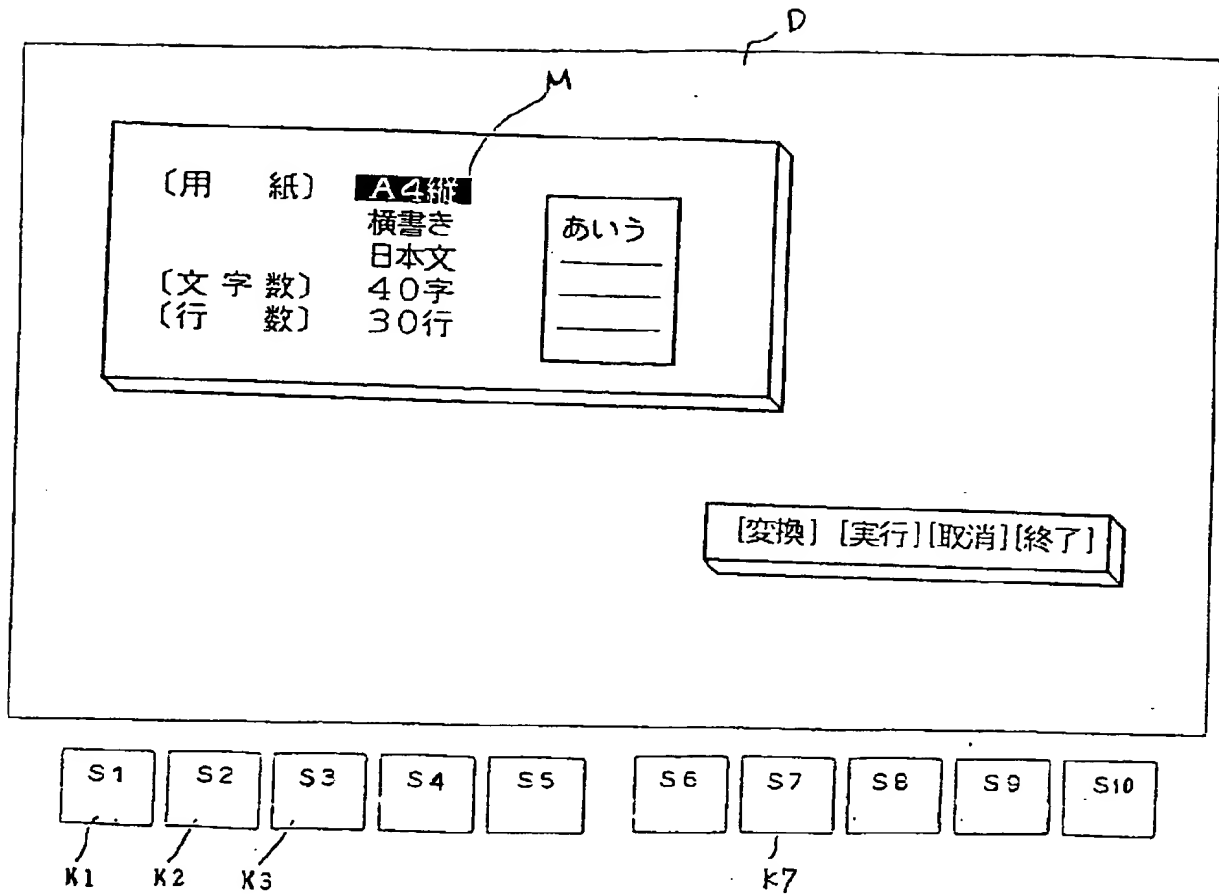
【図13】



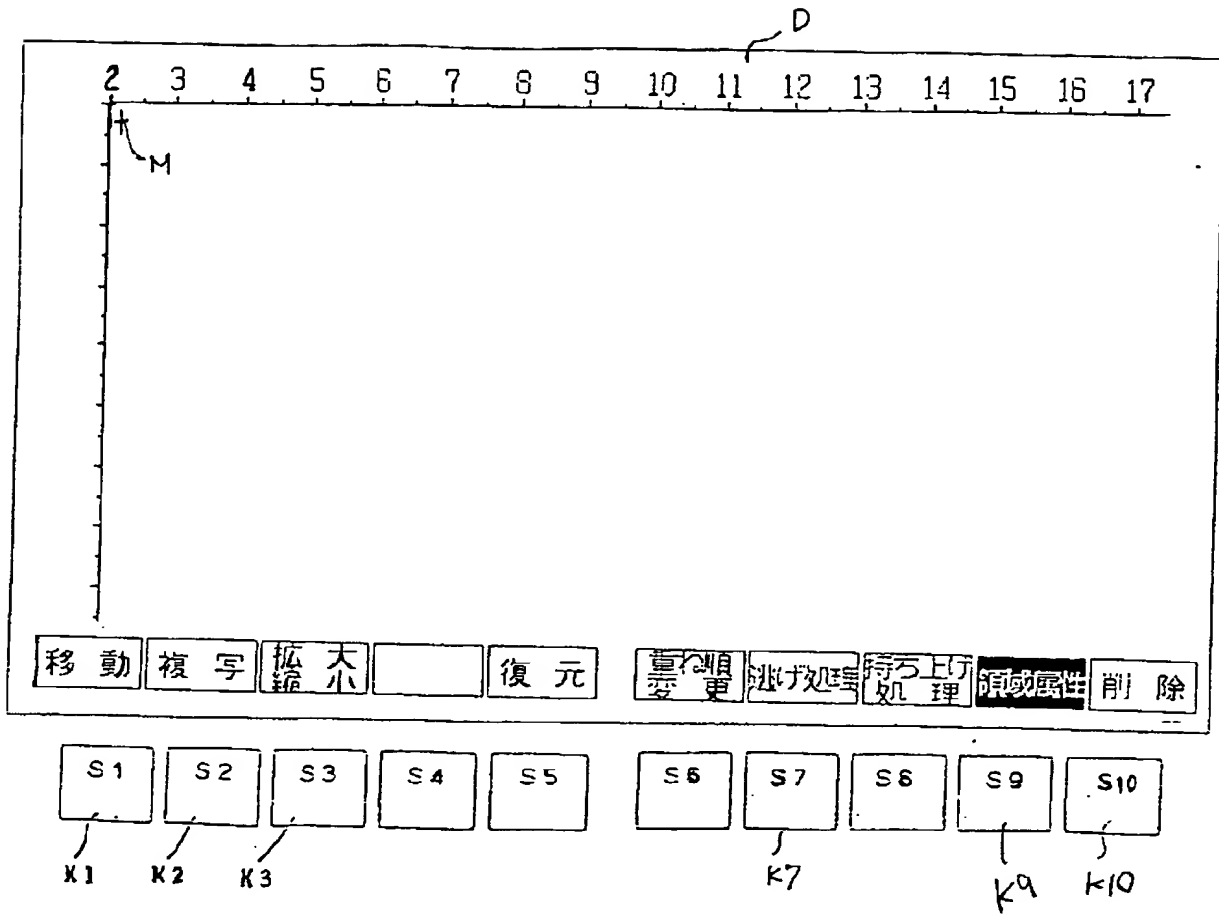
【図27】



【図14】

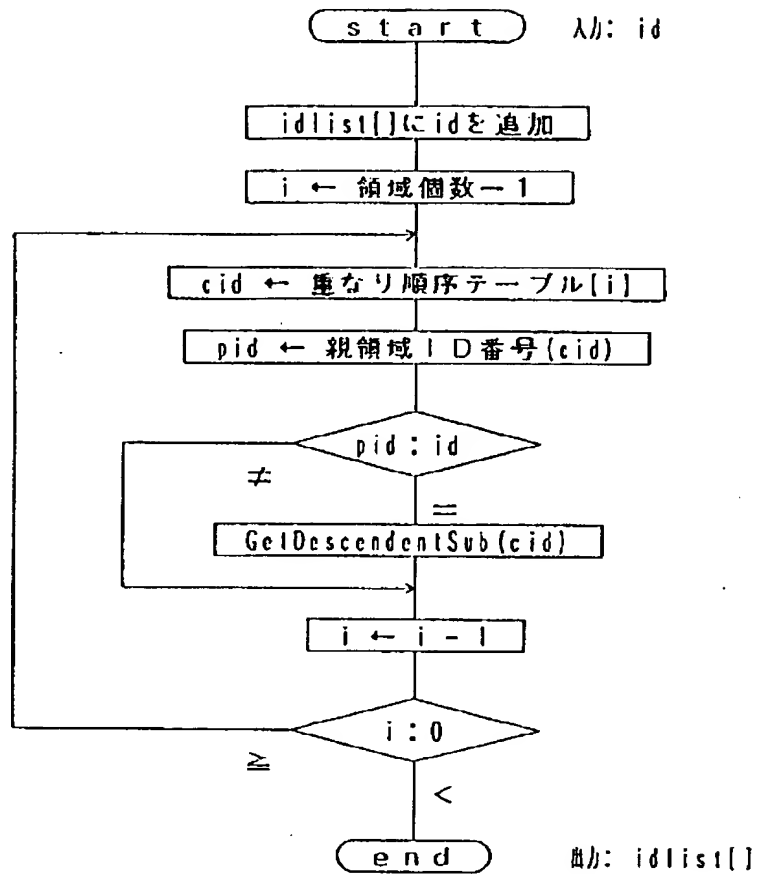


【図 15】



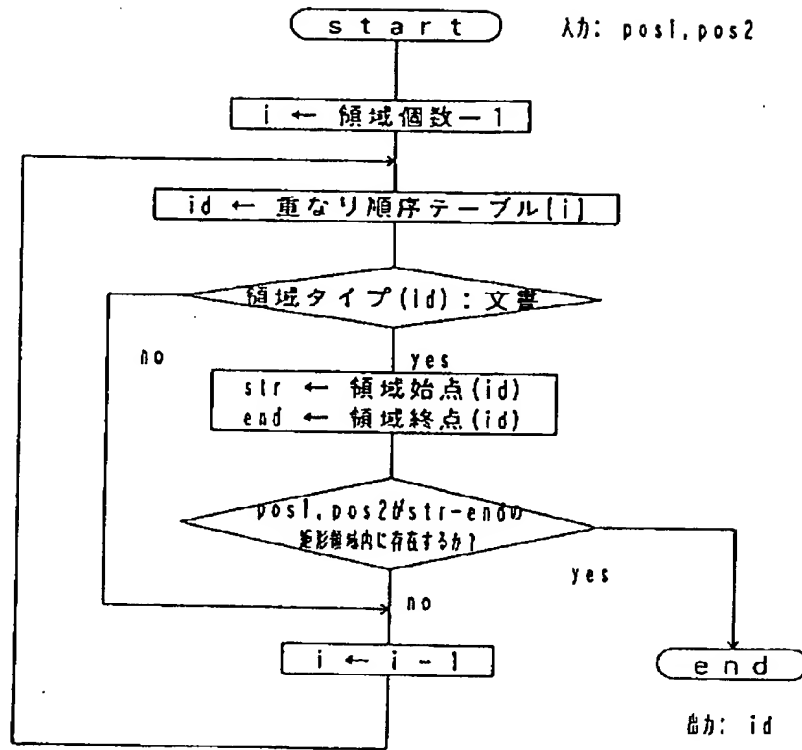


【図18】

**GetDescendentSub**

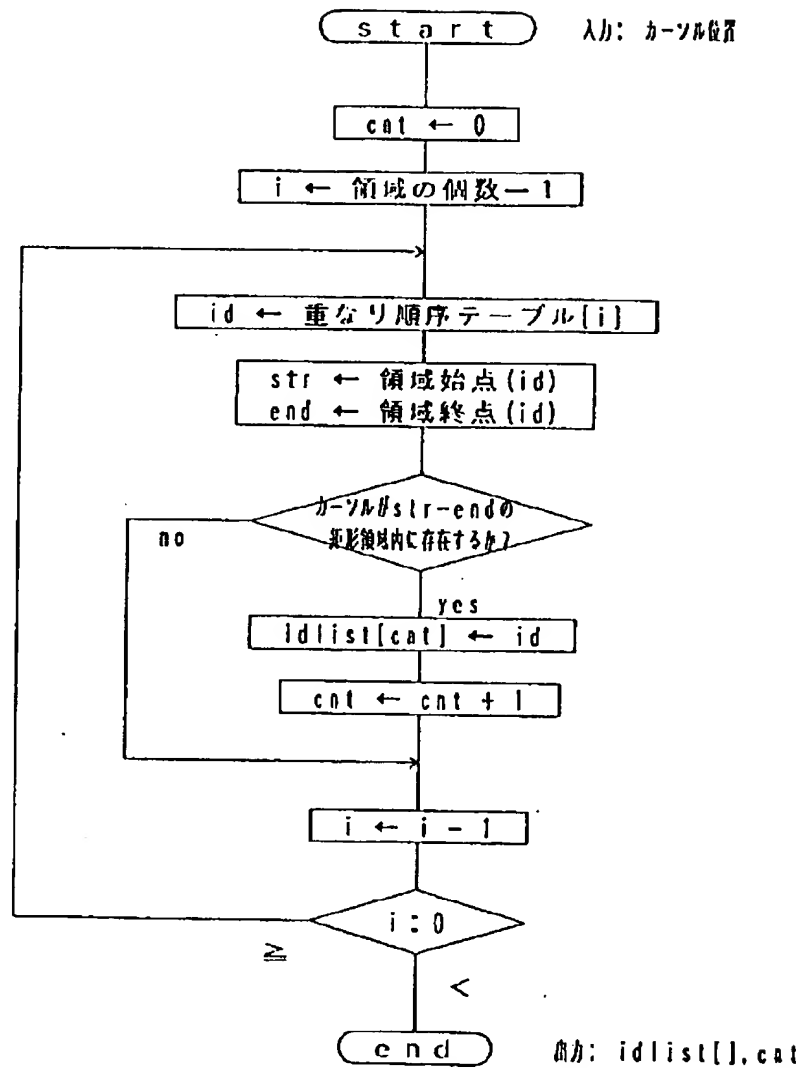
【図19】

GetParentId

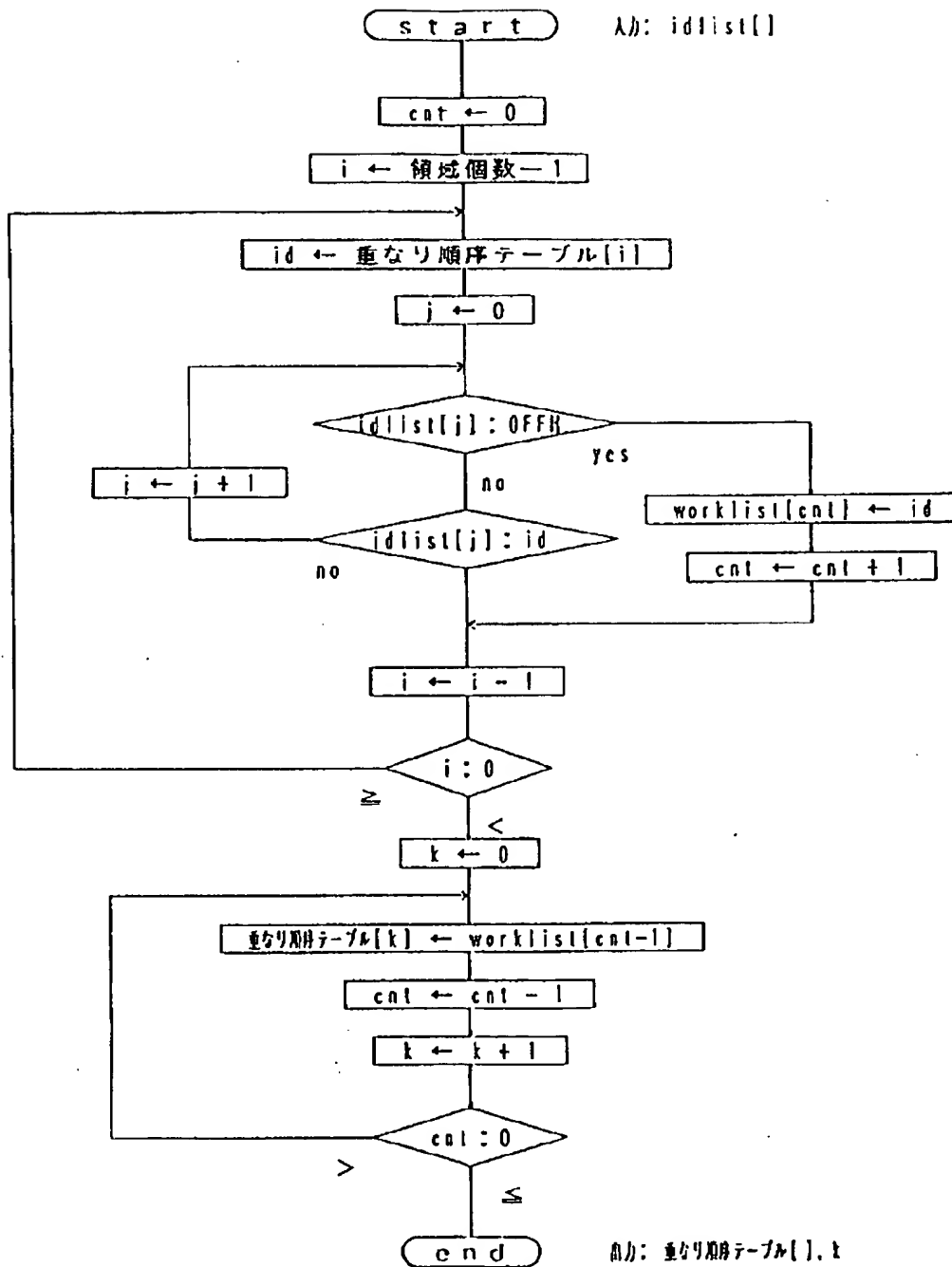


【図20】

GetHighest

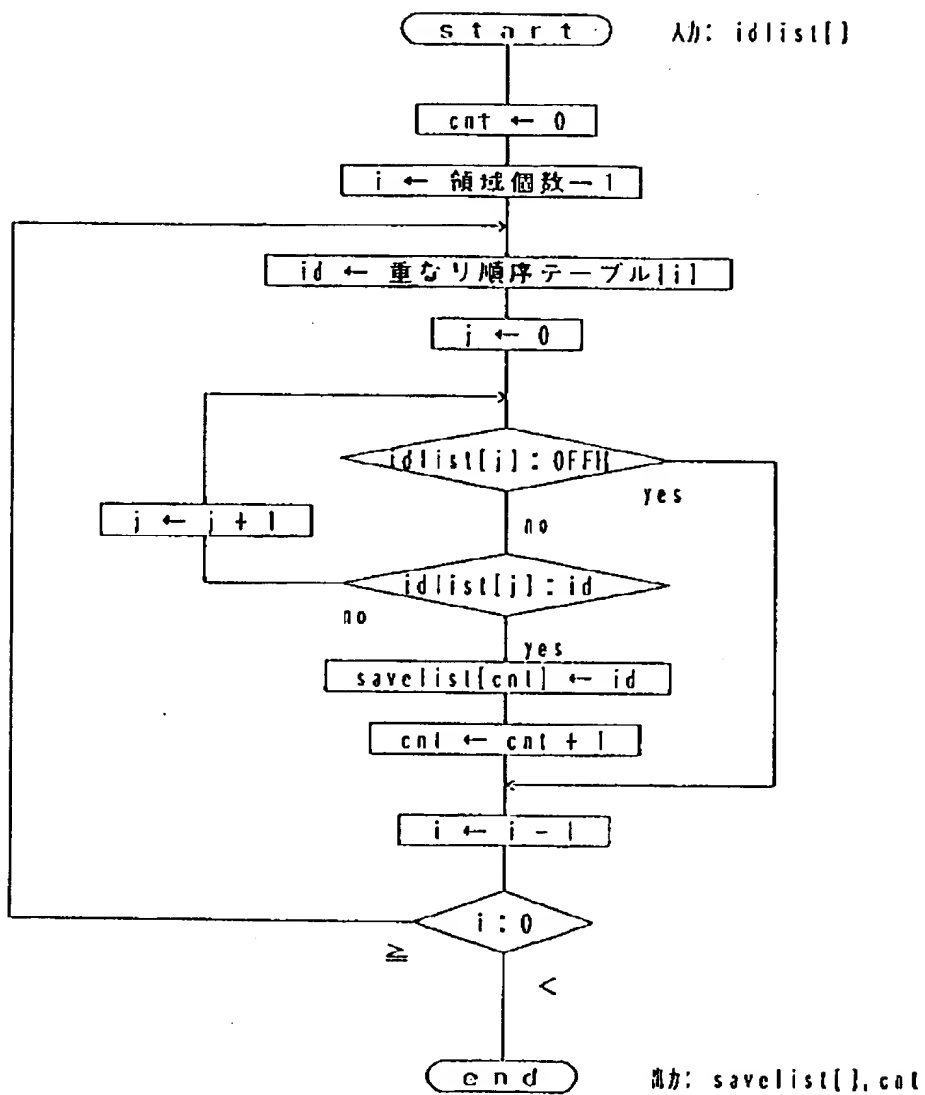


【図21】

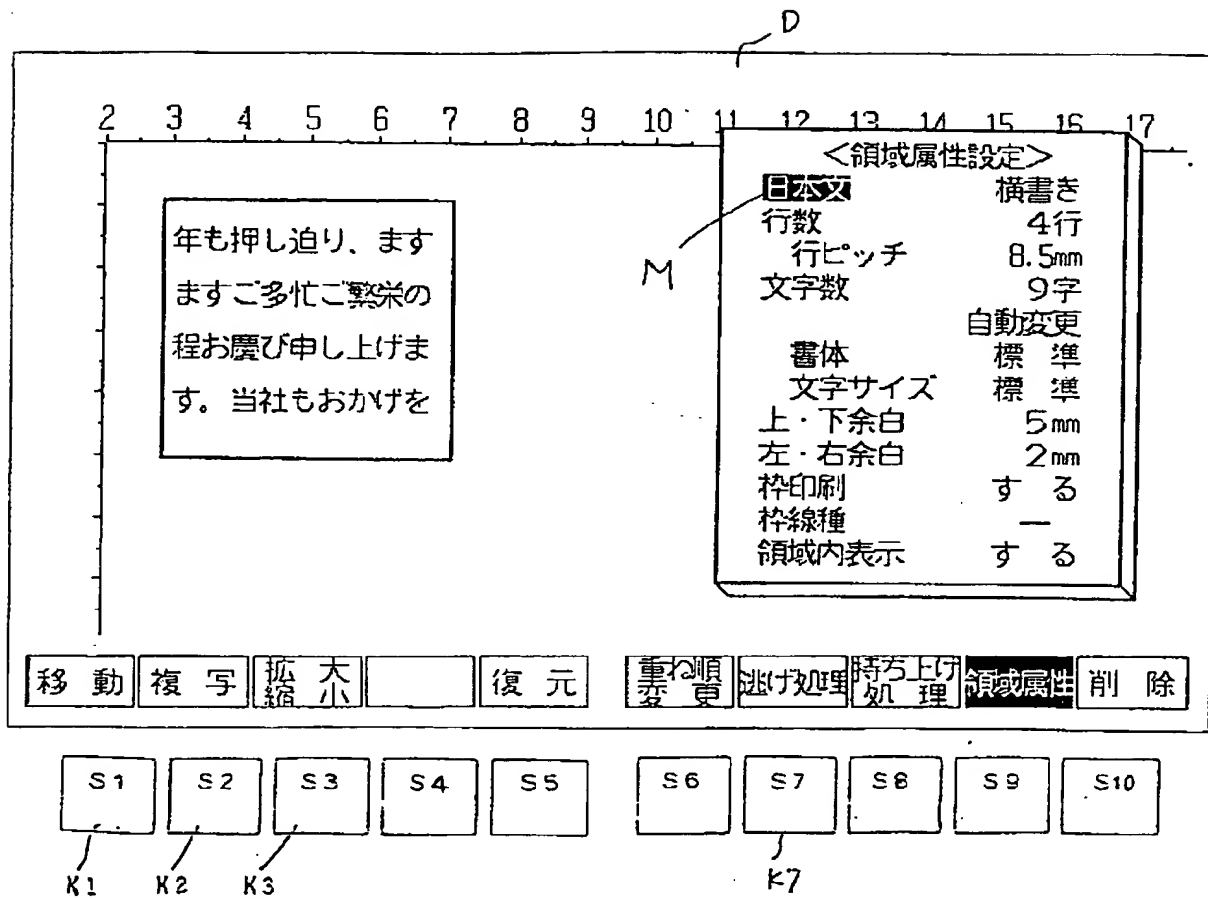


【図22】

SaveOverlap

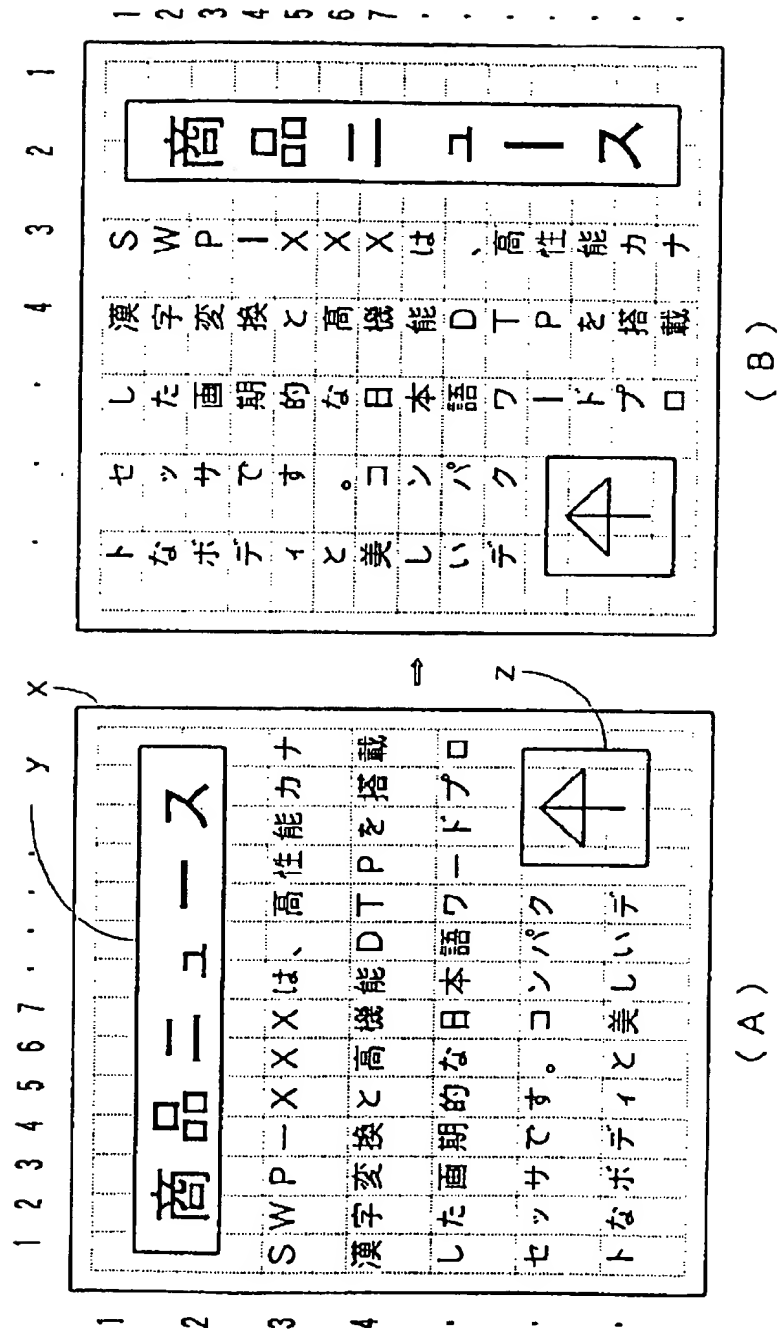


【図23】



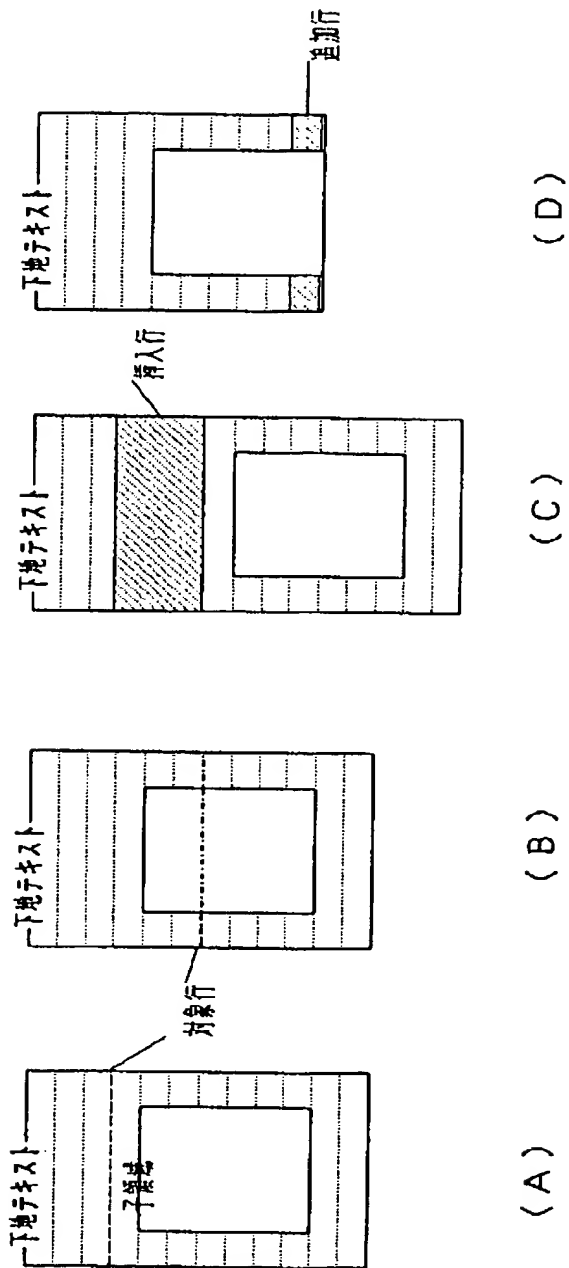


【図25】

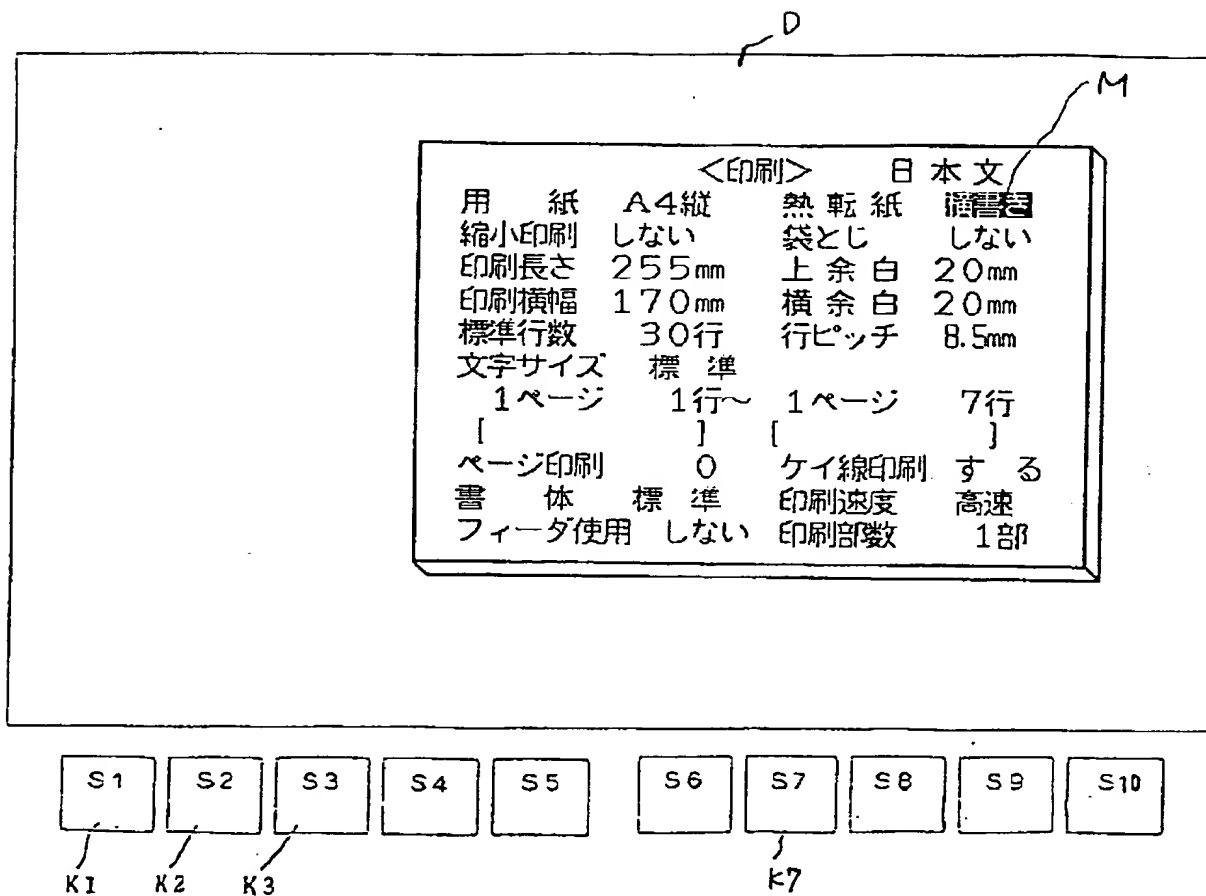




【図26】



【図28】



【図29】

# スーパータイトル

大垣市の三大祭りの一つ、「十萬石祭り」が一月一〇日に大垣駅前通り周辺で繰り広げられました。祭りのメインである「みこし行列」に三五企業・三八基のみこしを初め、町内界の子供みこし、大人みこし約一〇〇基が参加。沿道には約八万人（主催者調べ）が詰めかけ、祭り気分も最高潮。三洋電機はコーポレートスローガンを大きく掲げ、「わっしょい、わっしょい」と威勢よく躍り出しました。約四〇名の田舎手（写真下）の皆さん、ご苦労様でした。大垣市の三大祭りの一つ、「十萬石祭り」が一月一〇日に大垣駅前通

り周辺で繰り広げられました。祭りのメインである「みこし行列」に三五企業・三八基のみこしを初め、町内界の子供みこし、大人みこし約一〇〇基が参加。沿道には約八万人（主催者調べ）が詰めかけ、祭り気分も最高潮。三洋電機はコーポレートスローガンを大きく掲げ、「わっしょい、わっしょい」と威勢よく躍り出しました。約四〇名の田舎手（写真下）の皆さん、ご苦労様でした。大垣市の三大祭りの

一つ、「十萬石祭り」が一月一〇日に大垣駅前通り周辺で繰り広げられました。祭りのメインである「みこし行列」に三五企業・三八基のみこしを初め、町内界の子供みこし、大人みこし約一〇〇基が参加。沿道には約八万人（主催者調べ）が詰めかけ、祭り気分も最高潮。三洋電機はコーポレートスローガンを大きく掲げ、「わっしょい、わっしょい」と威勢よく躍り出しました。約四〇名の田舎手（写真下）の皆さん、ご苦労様でした。大垣市の三大祭りの

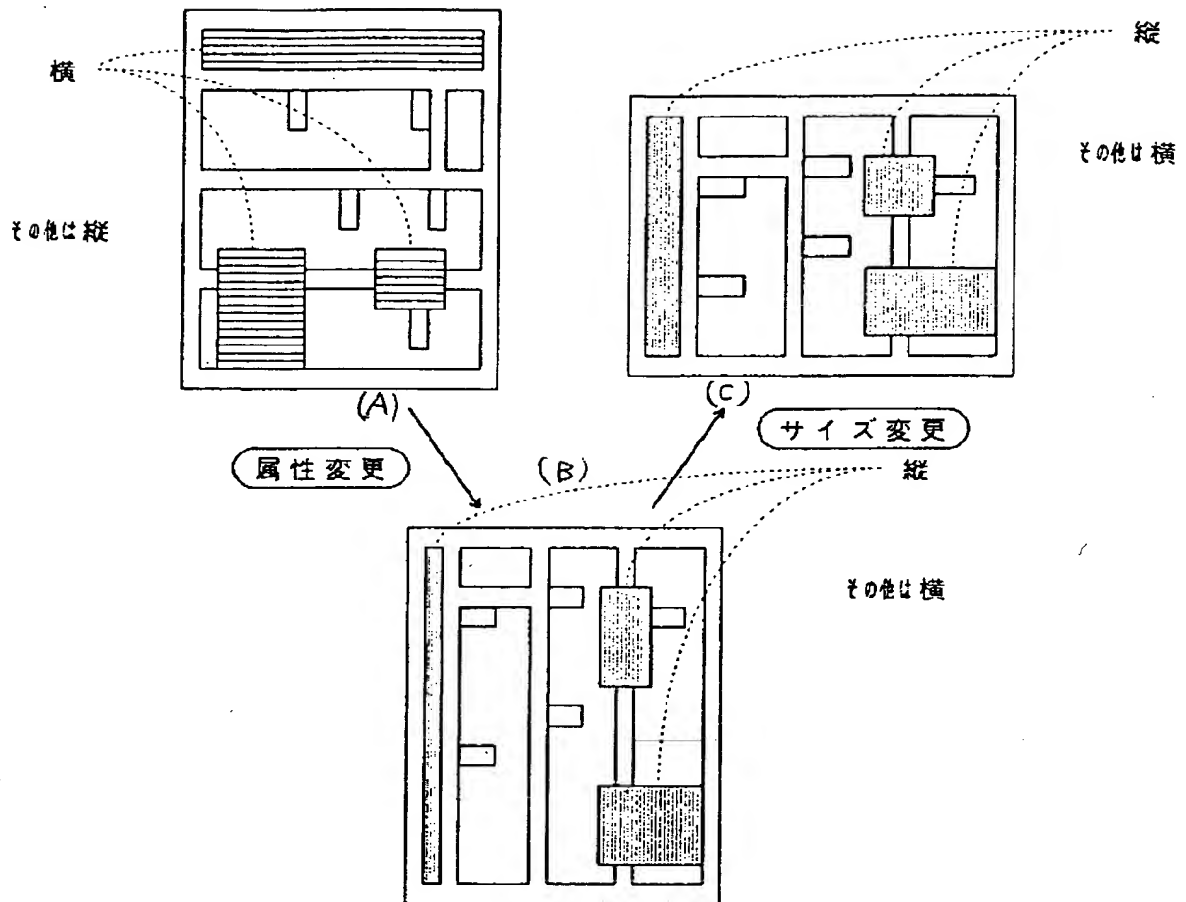
に決定した。建物の高さは現在と同じで、建物の幅もほぼ同じです。予定は、完成は、予定です。

大垣市の三大祭りの一つ、「十萬石祭り」が一月一〇日に大垣駅前通り周辺で繰り広げられました。祭りのメインである「みこし行列」に三五企業・三八基のみこしを初め、町内界の子供みこし、大人みこし約一〇〇基が参加。沿道には約八万人（主催者調べ）が詰めかけ、祭り気分も最高潮。三洋電機はコーポレートスローガンを大きく掲げ、「わっしょい、わっしょい」と威勢よく躍り出しました。約四〇名の田舎手（写真下）の皆さん、ご苦労様でした。大垣市の三大祭りの

【図30】

スーパータイトル	
<p>大連市の三大祭りの一つ、「十萬石祭り」が、一〇月一〇日に大連駅前通り周辺で繰り広げられました。祭りのメインである「みこし行列」に三三三企業・三八五のみなしを初め、市内各界の子ども、大人、若者が参加。沿道には約八万人（三連寺調べ）が詰めかけ、祭り気分も最盛期。三連寺電線はコーポレートスローガンを大きく掲げ、「わっしょい、わっしょい」と威勢よく繰り出しました。約四〇名の若き手（写真下）の皆さん、ご苦労様でした。</p> <p>大連市の三大祭りの一つ、「十萬石祭り」が一〇月一〇日に大連駅前通り</p>	
<p>大連市の三大祭りの一つ、「十萬石祭り」が、一〇月一〇日に大連駅前通り周辺で繰り広げられました。祭りのメインである「みこし行列」に三三三企業・三八五のみなしを初め、市内各界の子ども、大人、若者が参加。沿道には約八万人（三連寺調べ）が詰めかけ、祭り気分も最盛期。三連寺電線はコーポレートスローガンを大きく掲げ、「わっしょい、わっしょい」と威勢よく繰り出しました。約四〇名の若き手（写真下）の皆さん、ご苦労様でした。</p> <p>大連市の三大祭りの一つ、「十萬石祭り」が一〇月一〇日に大連駅前通り</p>	<p>り周辺で繰り広げられました。祭りのメインである「みこし行列」に三三三企業・三八五のみなしを初め、市内各界の子ども、大人、若者が参加。沿道には約八万人（三連寺調べ）が詰めかけ、祭り気分も最盛期。三連寺電線はコーポレートスローガンを大きく掲げ、「わっしょい、わっしょい」と威勢よく繰り出しました。約四〇名の若き手（写真下）の皆さん、ご苦労様でした。</p> <p>大連市の三大祭りの一つ、「十萬石祭り」が一〇月一〇日に大連駅前通り</p>
<p>り周辺で繰り広げられました。祭りのメインである「みこし行列」に三三三企業・三八五のみなしを初め、市内各界の子ども、大人、若者が参加。沿道には約八万人（三連寺調べ）が詰めかけ、祭り気分も最盛期。三連寺電線はコーポレートスローガンを大きく掲げ、「わっしょい、わっしょい」と威勢よく繰り出しました。約四〇名の若き手（写真下）の皆さん、ご苦労様でした。</p> <p>大連市の三大祭りの一つ、「十萬石祭り」が一〇月一〇日に大連駅前通り</p>	<p>一つ、「十萬石祭り」が一〇月一〇日に大連駅前通り周辺で繰り広げられました。祭りのメインである「みこし行列」に三三三企業・三八五のみなしを初め、市内各界の子ども、大人、若者が参加。沿道には約八万人（三連寺調べ）が詰めかけ、祭り気分も最盛期。三連寺電線はコーポレートスローガンを大きく掲げ、「わっしょい、わっしょい」と威勢よく繰り出しました。約四〇名の若き手（写真下）の皆さん、ご苦労様でした。</p> <p>大連市の三大祭りの一つ、「十萬石祭り」が一〇月一〇日に大連駅前通り</p>

【図31】



フロントページの続き

(72)発明者 竹山 哲夫  
大阪府守口市京阪本通2丁目18番地 三洋  
電機株式会社内

(72)発明者 黒川 敏  
大阪府守口市京阪本通2丁目18番地 三洋  
電機株式会社内